Amendments to the Claims:

Rewrite the claims as set forth below. This listing of claims replaces all prior versions and listings of claims in the application:

1.   (currently amended) A method for nested control flow, the method comprising:

setting a context bit to at least one of: a first state and a ~~second state~~ second state:

receiving a first instruction having a plurality of extra bits;

~~determining whether to read~~ reading the context bit based on the plurality of extra bits; and

~~if the context bit is read, executing~~ executing the instruction when the context bit is in the first state.

2.   (currently amended) The method of claim 1 further comprising:

~~upon the executing of the first instruction, maintaining~~ maintaining a counter value wherein the counter value indicates a nesting depth of context bits that are set to a second state.

3.   (currently amended) The method of claim 2 wherein the counter value is stored in a non-dedicated memory device.

4.   (original) The method of claim 3 wherein the non-dedicated memory device is a general purpose register.

5.   (currently amended) The method of claim 2 further comprising:

prior to setting the context ~~bit~~ bits, resetting the counter value.

6.   (currently amended) The method of claim 2 further comprising:

receiving a second instruction having a plurality of extra bits;

~~determining whether to read~~ reading the context bit based on the plurality of extra bits;

~~if the context bit is read, executing~~ executing the second instruction when the context bit is in the first state; and

~~upon executing the second instruction, maintaining~~ maintaining the counter value.

7.   (currently amended) The method of claim 6 further comprising:

~~terminating a computation process using the counter value~~ exiting a nested control flow using the counter value.

8.   (currently amended) An apparatus for nested control flow, the apparatus comprising:

a processor having a context bit memory device capable of storing a context bit;

a first memory device storing a plurality of instructions, wherein each of the plurality of instructions includes a plurality of extra bits, and wherein the processor is operative to execute the plurality of instructions; and

a second memory device operably coupled to the processor, and wherein the second memory device ~~receiving~~ receives an incrementing counter instruction upon the execution of one of the plurality of instructions.

9.   (original) The apparatus of claim 8 further comprising:

a context bit memory device capable of storing the context bit.

10.   (original) The apparatus of claim 8 wherein the second memory device is a general purpose register.

11.   (currently amended) The apparatus of claim 8 wherein the processor:

receives a first instruction having a plurality of extra bits from the ~~first~~ memory ~~and the processor~~ device; and

~~determines whether to read~~ reads the context bit based on the plurality of extra bits.

12.   (currently amended) The apparatus of claim 11 wherein the processor:

executes the first instruction when the context bit is read and is in a first ~~state and the processor~~ state; and

~~maintaining~~ maintains a counter value wherein the counter value indicates a nesting depth of context bits that are set to a second state, ~~in response to~~ using the incrementing counter instruction.

13.   (currently amended) The apparatus of claim 12 wherein the ~~process receives~~ processor:

receives a second instruction having a plurality of extra bits from the ~~memory and the processor determines~~ first memory device;

~~whether to read~~ reads the context bit based on the plurality of extra ~~bits and if~~ bits;

~~the context bit is read, executing~~ executes the second instruction when the context bit is in a first ~~state and incrementing~~ state; and

increments the counter ~~bit in response to~~ value using the incrementing counter instruction.

14.   (currently amended) A graphics processing device comprising:

a plurality of arithmetic logic units, each of the plurality of arithmetic logic units having a context bit memory device capable of storing a context bit;

a first memory device storing a plurality of instructions, wherein each of the plurality of instructions includes a plurality of extra bits, <u>and wherein</u> the arithmetic logic units <u>are</u> operative to execute the plurality of instructions; and

a second memory device operably coupled to the processor, <u>wherein</u> the second memory device ~~receiving~~ <u>receives</u> an incrementing counter instruction upon the execution of one of the plurality of instructions.

15.   (original) The graphics processing device of claim 14 wherein the second memory device is a general purpose register.

16.   (currently amended) The graphics processing device of claim 14 wherein each of the plurality of arithmetic logic units:

receive at least one of the plurality of ~~instructions and the arithmetic logic units determine~~ <u>instructions; and</u>

~~whether to read~~ <u>reads</u> the context bit based on the plurality of extra bits.

17.   (currently amended) The graphics processing device of claim 16 wherein the plurality of arithmetic logic units:

execute the instructions when the context bit is read and is in a first ~~state and maintaining~~ <u>state; and</u>

<u>maintains</u> a counter value wherein the counter value indicates a nesting depth of context bits that are set to a second state, ~~in response to~~ <u>using</u> the incrementing counter instruction.

18.   (currently amended) The graphics processing device of claim 17 wherein the plurality of arithmetic logic units are operative to ~~terminate a processing chain utilizing the~~

~~context bit stored in the second memory device~~ exit a nested control flow using the context bit stored in the second memory device.

19. (currently amended) A method for nested control flow, the method comprising:

setting a context bit to at least one of: a first state and a second ~~state~~ state;

receiving a first instruction having a plurality of extra bits;

~~determining whether to read~~ reading the context bit based on the plurality of extra bits;

~~if the context bit is read,~~ executing the first instruction when the context bit is in the first state; and

upon the executing of the first instruction, maintaining a counter value wherein the counter value indicates a nesting depth of context bits that are set to a second ~~state.~~ state in a general purpose register.

20. (currently amended) The method of claim 19 further comprising:

receiving a second instruction having a plurality of extra bits;

~~determining whether to read~~ reading the context bit based on the plurality of extra bits;

~~if the context bit is read,~~ executing the second instruction when the context bit is in the first state; and

~~upon executing the second instruction,~~ incrementing the counter value.

21. (currently amended) The method of claim 20 further comprising:

~~terminating a computation processing using the counter value~~ exiting a nested control flow using the counter value.